AF/2186 IFW

Attorney's Docket No.: 10559/364001/P8247X

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant :	David I. Poisner	Art Unit :	2186
Serial No.:	09/672,345	Examiner :	Hong Chong Kim
Filed :	September 28, 2000		
Title :	ACCESSING MULTI-PORTED MEMORY		

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

BRIEF ON APPEAL

(1) Real Party in Interest

This case is assigned of record to Intel Corporation, who is hence the real party in interest.

(2) Related Appeals and Interferences

There are no known related appeals and/or interferences.

(3) Status of Claims

Claims 34-50 are currently pending in the application. Claims 34-50 stand rejected and are being appealed.

(4) Status of Amendments

No amendment has been filed after final rejection.

CERTIFICATE OF MAILING BY FIRST CLASS MAIL

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail with sufficient postage on the date indicated below and is addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

November 30, 2004
Date of Deposit

R-R Ippolito
Signature

Roxanne Ippolito
Typed or Printed Name of Person Signing Certificate

(5) Summary of Claimed Subject Matter

The present application teaches systems and techniques that may be used to improve the efficiency of information transfer using a multi-ported memory. The efficiency may be obtained by directing control/status accesses to a multi-ported memory such as memory 18 of Fig. 1, rather than main memory 22 of Fig. 1. (Please see page 4, lines 3-8 of the specification).

By directing control/status accesses to the multi-ported memory, performance degradation due to bus snooping may be avoided for those accesses. "Bus snooping" refers to monitoring the communication channels for write accesses to main memory. (Please see page 3, lines 16-19 of the specification). Although bus snooping helps ensure cache consistency, it can increase the cost and decrease the performance of the microprocessor. (Please see page 3, lines 19-22 of the specification).

The independent claims being appealed are claims 34, 45, and 48. The features of independent claim 34 are shown in Figs. 1 to 3, and exemplary embodiments are described in the specification as follows.

Claim 35 recites a processor (e.g., CPU 12), a main memory (e.g., main memory 22), and a multi-ported memory in communication with the processor and the main memory (e.g., dual-ported memory 18).

The multi-ported memory has a storage capacity of about 4 kilobytes or greater (see, e.g., Fig. 3 and page 8, lines 8-21 of the specification).

Claim 35 further recites that the system is configured to receive a request to write information to a memory location, wherein the information has an information type equal to data or control information, and wherein the system is further

configured to determine a memory destination between the main memory or the multi-ported memory based on the information type.

For example, control/status accesses are directed principally to dual-ported memory 52, rather than to main memory 46, thereby avoiding a bus snoop and its attendant delay (please see page 7, lines 3-6 of the specification). Note that the term "accesses" refers to reading or writing data (please see page 1, lines 12-13 of the specification).

Accesses may be directed in a number of ways. For example, a computer's operating system may direct appropriate accesses to dual-ported memory 18. (Please see page 4, lines 10-12 of the specification). Another way is for components to be configured to look for the multi-ported memory; for example, an input/output device may check to see whether the multi-ported memory is available, and if so may direct accesses to the multi-ported memory. If not, the device may access main memory. (Please see page 4, lines 12-19).

Similarly, claims 45 (method) and 48 (Beauregard) include the features receiving a request to write information to a memory location (e.g., a control/status access or a data access), determining an information type equal to data or control information for the information, and determining a memory destination between a main memory and a multi-ported memory based on the information type, the multi-ported memory having a storage capacity of about 4 kilobytes or greater.

(6) Grounds of Rejection

Claims 34-40 and 43-50 stand rejected under 35 U.S.C. 103(a) as allegedly being unpatentable over U.S. Patent No. 4,371,932 to Dinwiddie, Jr. et al. ("Dinwiddie") in view of U.S. Patent No. 6,680,908 to Gibson et al. ("Gibson"). Claim 41

stands rejected under 35 U.S.C. 103(a) as allegedly being unpatentable over Dinwiddie and Gibson, and further in view of U.S. Patent No. 5,784,699 to McMahon et al. ("McMahon"). Claim 42 stands rejected under 35 U.S.C. 103(a) as allegedly being unpatentable over Dinwiddie and Gibson, and further in view of U.S. Patent No. 5,546,554 to Young et al. ("Young").

(7) Argument

Claim 34

Claim 34 is patentable over the combination of Dinwiddie and Gibson because there is no motivation to increase the storage capacity of elements 25, 27, or 30 of Dinwiddie to be about 4 kilobytes or greater.

In the final office action having a mailing date of July 7, 2004 ("the final office action"), it is acknowledged that Dinwiddie does not disclose that elements 25, 27, and 30 have a storage capacity of greater than 4 kilobytes or greater. However, the office action alleges that it would have been obvious to incorporate a storage capacity of about 4 kilobytes or greater into the invention of Dinwiddie. (Please see page 3 of the final office action). The final office action alleges that the motivation for the modification would be for the purpose of providing adequate space for the system, thereby preventing system slow down or crash. (Please see page 3 of the final office action).

However, such a modification of Dinwiddie with the teachings of Gibson would not have been obvious. Although increasing the size of the buffer of Gibson may prevent system slow down or crash, the same benefit does not accrue when the storage capacity of elements 25, 27, or 30 is increased. Unlike the buffer of Gibson, elements 25, 27, and 30 are each elements

that provide minimal storage, for a particular purpose. There is no motivation to increase the storage capacity of these elements to be about 4 kilobytes or greater.

For example, element 25 is a command register file. (Please see Figure 1 of Dinwiddie). The "particular purpose" of element 25 is to store input/output commands for each of the input/output devices. (Please see column 15, lines 3-18 of Dinwiddie).

The specification gives an example of command register file 25 at column 14, lines 50-60 (emphasis added):

> For sake of example, it is assumed that command register file 25 is comprised of two Texas Instruments type SN74LS670 4-by-4 register files. These register files are manufactured and marketed by Texas Instruments, Inc., of Dallas, Tex. These two 4-bit wide register files are operated in unison to provide, in effect, a single register file having a width of 8 bits or 1 byte, with four one-byte locations being separately addressable. In other words, register file 25 <u>is just like a stack of four addressable 1-byte registers</u>.

Thus, in this example from the specification, the storage capacity of register file 25 is <u>four bytes</u>, corresponding to four input/output devices.

Similarly, the particular purpose of cycle steal address register 27 is to store a particular address. The specification describes cycle steal address register 27 at column 13, lines 51-53:

> The cycle steal address register 27 is a 16-bit register and, for example, may be comprised of a pair of Intel 8282 8-bit input/output port units.

Thus, in this example from the specification, the storage capacity of cycle steal address register 27 is two bytes.

Dinwiddie does not give an exemplary storage capacity for handshake, interrupt and miscellaneous controls 30. However, Figs. 12A and 12B of Dinwiddie show interrupt and cycle steal handshaking unit 33 of Fig. 2C of Dinwiddie. As noted in column 8, lines 24-26, unit 33 of Fig. 2C implements both the interrupt and cycle steal handshaking functions of the control unit 30 of Fig. 1. As one would expect, unit 33 of Dinwiddie does not include elements with a storage capacity of about four kilobytes or greater. Instead, unit 33 includes logic elements, registers, and the like.

As noted above, increasing the storage capacity of any of elements 25, 27, or 30 of Dinwiddie would not provide the purported benefit. The capacity of these elements is selected based on the function of the element. For example, the capacity of element 25 is based on the number of input/output devices. The capacity of element 27 is based on the size of the address information. The capacity of element 30 is based on the information needed to provide the described control functions. Certainly there is no motivation to increase the storage capacity by <u>three orders of magnitude</u> as the rejection postulates.

In the response to arguments section of the final office action, it is alleged that since there is some teaching in Dinwiddie that these elements may include multiple registers, their modification to provide the claimed storage capacity would have been obvious. (Please see page 8 of the final office action).

For example, the final office action alleges that Fig. 8 of Dinwiddie discloses a register array with a storage capacity of multiples of 32 bits. However, Fig. 8 neither teaches nor suggests that the storage capacity of elements 25, 27, or 30 be

increased to about 4 kilobytes or greater. First, Fig. 8 shows the internal construction of a microprocessor, rather than showing any of the elements 25, 27, or 30. Second, Fig. 8 illustrates a register array with a small number of registers. Assuming each were 32 bits (4 bytes), it would take 1000 registers to increase the storage capacity of the register array to 4 kilobytes.

The office action further alleges that the term "file," which is used to refer to element 25, usually refers to a storage capacity of much greater than 32 bits. However, as noted above, the exemplary command register file 25 has a storage capacity of four bytes. Additionally, the office action alleges that the term "controls" generally refers to multiple registers. Although control elements may include multiple registers, it would take 1000 32 bit registers to have the claimed storage capacity.

Claim 34 is thus patentable over the combination of Dinwiddie and Gibson because there is no motivation to increase the storage capacity of elements 25, 27, or 30 of Dinwiddie to be about 4 kilobytes or greater.

Claims 35-44

Claims 35-44 depend from claim 34, and are thus patentable for at least the same reasons as stated above with respect to claim 34.

Claim 36

Claim 36 is patentable for at least the additional reason that the combination of Dinwiddie and Gibson neither teaches nor suggests a peripheral device controller configured to "determine the memory destination based on the information type," as recited in claim 36.

The office action identifies peripheral device controllers 17-20 of Fig. 1 of Dinwiddie as teaching this feature. However, Dinwiddie clearly teaches that all data transfers from the peripheral devices of Dinwiddie are made via dual port storage unit 22 of Dinwiddie. (Please see the Abstract of Dinwiddie). Thus, regardless of the information type (data or control information), the peripheral device controllers transfer the data to dual port storage unit 22. Thus, the peripheral device controllers of Dinwiddie cannot be said to determine a memory destination based on the information type.

For at least this additional reason, claim 36 is patentable over the combination of Dinwiddie and Gibson.

Claim 41

Claim 41 is patentable for at least the additional reason that there is no motivation to modify Dinwiddie to modify any of elements 25, 27, or 30 to include reservation bits mapped to blocks of general-purpose memory in the multi-ported memory, as recited in claim 41.

The office action alleges that the motivation is to provide fast search and allocation/deallocation of availability of a block, citing column 3, lines 7-26 of McMahon. Again, although this benefit may be obtained in the dynamic memory allocation system of McMahon, it would not accrue if elements 25, 27, or 30 of Dinwiddie were modified to include reservation bits. Elements 25, 27, or 30 are not related to searching or allocation/deallocation of memory blocks.

Indeed, it is not clear how such a modification would be made. As noted above, each of elements 25, 27, and 30 of Dinwiddie serves a particular purpose in the system of Dinwiddie, unrelated to memory allocation.

For at least this additional reason, claim 41 is patentable over the combination of Dinwiddie, Gibson, and McMahon.

Claims 45-50

Independent claims 45 and 48 include features similar to those discussed above with respect to claim 34, and are thus patentable for similar reasons. Claims 46 and 47, and 49 and 50 depend from claims 45 and 48 respectively, and are thus patentable for the same reasons.

The brief fee of $340 is enclosed. Please apply any other charges or credits to Deposit Account No. 06-1050.

Respectfully submitted,

Date: 11/30/04

Linda G. Gunderson
Reg. No. 46,341

Fish & Richardson P.C.
Customer Number: 20985
12390 El Camino Real
San Diego, California 92130
Telephone: (858) 678-5070
Facsimile: (858) 678-5099

10459563.doc

Appendix of Claims

34. A data processing system, comprising:

a processor;

a main memory;

a multi-ported memory in communication with the processor and the main memory, the multi-ported memory having a storage capacity of about 4 kilobytes or greater; and

wherein the system is configured to receive a request to write information to a memory location, wherein the information has an information type equal to data or control information, and wherein the system is further configured to determine a memory destination between the main memory or the multi-ported memory based on the information type.

35. The system of claim 34, further comprising an operating system configured to determine the memory destination based on the information type.

36. The system of claim 34, wherein the system further includes:

a peripheral device; and

a peripheral device controller, wherein the controller is configured to determine the memory destination based on the information type.

37. The system of claim 34, wherein the multi-ported memory is included in a memory controller.

38. The system of claim 34, wherein the multi-ported memory is dual-ported.

39. The system of claim 34, wherein the multi-ported memory and memory controller are integrated into a single chip.

40. The system of claim 34, wherein the multi-ported memory includes memory chosen from the group consisting of static random access memory and dynamic random access memory.

41. The system of claim 34, wherein the multi-ported memory stores reservation bits mapped to blocks of general-purpose memory in the multi-ported memory.

42. The system of claim 34, wherein virtual addresses within multi-ported memory are mapped to physical addresses with smart addressing.

43. The system of claim 34, further including:

a memory controller in communication with the main memory and the multi-ported memory; and

a peripheral device in communication with the memory controller via an input/output bus.

44. The system of claim 34, wherein for information with an information type equal to control information, the system is configured to determine the memory destination to be the multi-ported memory and not the main memory.

45. A method comprising:

receiving a request to write information to a memory location;

determining an information type equal to data or control information for the information; and

determining a memory destination between a main memory and a multi-ported memory based on the information type, the multi-

ported memory having a storage capacity of about 4 kilobytes or greater.

46. The method of claim 45, further comprising:

writing the information to the memory destination based on the determining the memory destination.

47. The method of claim 45, wherein determining the memory destination between the main memory and the multi-ported memory based on the information type comprises determining the memory destination to be the multi-ported memory for the information type equal to control information.

48. An article comprising a computer-readable medium which stores computer-executable instructions, the instructions causing one or more machines to perform operations comprising:

receiving a request to write information to a memory location;

determining an information type equal to data or control information for the information; and

determining a memory destination between a main memory and a multi-ported memory based on the information type, the multi-

ported memory having a storage capacity of about 4 kilobytes or greater.

49. The article of claim 48, further comprising:

writing the information to the memory destination based on the determining the memory destination.

50. The article of claim 48, wherein determining the memory destination between the main memory and the multi-ported memory based on the information type comprises determining the memory destination to be the multi-ported memory for the information type equal to control information.